A Method and Circuit for Measuring On-chip, Cycle-to-Cycle Clock Jitter.

FIELD OF THE INVENTION

[001] This invention relates generally to clocking circuits on integrated electronic circuits. More particularly, this invention relates to measuring on-chip, cycle-to-cycle clock jitter.

BACKGROUND OF THE INVENTION

[002] A microprocessor used in many large computer systems may include memory elements, combinational logic, and a clocking system. The memory elements may be arranged in sets, sometimes called registers that may correspond to the word size used in a computer system. Between at least some sets of memory elements are combinational logic circuits. At the end of a clock cycle, which is also the beginning of the next clock cycle, data on the output of the combinational logic circuitry is stored in a first set of memory elements. This data appears on the output of the set of memory elements, and therefore on the input of other combinational logic circuitry. The other logic circuitry performs the designed logic function on the data, and at the end of the clock cycle the output of this combinational logic is stored in a next set of memory elements. This process is repeated as the computer operates. In other words, data is processed by combinational logic circuity, stored in memory elements, and then passed on to other combinational logic circuitry. A system clock, often a PLL (Phase Locked Loop) controls the clocking of information from one state to the next state.

[003] On many microprocessors a high frequency clock signal is distributed across an entire die. As clock speeds exceed 2 GHz and die size exceeds 400 square

millimeters, clock distribution may become more difficult. An ideal PLL aligns the phase (edge time) of a clock signal arriving at a memory element to the system clock. However, this usually does not happen in practice. Long term jitter or variation of the phase alignment increases as the size of a die increases. As a result, the time allowed to propagate data from one memory element, through combinational logic, into another memory element is reduced. This time is often called a "clock budget."

[004] In addition to long term jitter, a PLL may produce cycle-to-cycle jitter. Cycle-to-cycle jitter is a measure of the variation in the clock cycle due to the PLL. Cycle-to cycle jitter may occur, among other things, due to temperature variation or changes in on-chip voltages. In order to reduce the probability of system errors, the clock budget should be reduced in order to compensate for cycle-to-cycle jitter.

[005] Measuring the jitter performance of microprocessors can be a difficult testing challenge. The cost of external equipment and testing time can be expensive. In addition, the process of probing a die in order to measure jitter usually destroys a die. Because a die is usually destroyed after one probing, additional jitter measurements at different locations on a die are usually not possible. There is a need in the art to reduce the cost of measuring jitter, reduce the number of die that are destroyed by probing, and make more than one jitter measurement on an individual die.

[006] An embodiment of this invention allows for on-die measurement of cycle-to-cycle jitter at multiple locations through out the clock distribution. These measurements can be made without probing the die thus reducing the number of die destroyed and reducing the cost of measuring jitter.

SUMMARY OF THE INVENTION

[007] In a preferred embodiment, the invention provides a method and circuit for measuring on-chip, cycle-to-cycle, jitter. Several copies of a circuit comprising a programmable delay line, a programmable phase comparator, and two counters are placed at different locations on an IC near a clock signal. The programmable delay line creates a clock signal that is delayed by one clock cycle. This delayed clock signal is compared in time to the clock signal by the programmable phase comparator. If the difference in time between the delayed clock signal and the clock signal is greater than the dead time programmed in the programmable phase comparator, the first counter is triggered. If the difference in time is negative and the absolute value is greater than the dead time, the second counter is triggered. A statistical distribution, based on the values of the counters, is created. This distribution is used to predict on-chip, cycle-to-cycle jitter.

[008] Other aspects and advantages of the present invention will become apparent from the following detailed description, taken in conjunction with the accompanying drawings, illustrating by way of example the principles of the invention.

BRIEF DESCRIPTION OF THE DRAWINGS

[009] Figure 1 is a schematic drawing of a programmable delay line.

[010] Figure 2 is a block diagram of a circuit for measuring jitter.

[011] Figure 3 is a drawing illustrating how individual jitter measuring circuits may be placed at different locations on an integrated circuit.

[012] Figure 4 is a drawing showing distributions outside the dead zone at various programmed values of the dead zone.

[013] Figure 5 is a statistical distribution illustrating how jitter measurement made on-chip may be used to predict cycle-to-cycle jitter.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

[014] Figure 1 is a schematic drawing of a programmable delay line. The input, **100**, to the programmable delay line is connected to the input of inverter **INV1**. The output, **102**, of inverter **INV1** is connected to the input, **102**, of inverter **INV2**. The output, **104**, of inverter **INV2** is connected to the input, **104**, of inverter **INV3**. The output, **106**, of inverter **INV3** is connected to the input, **106**, of inverter **INV4**. The output, **108**, of inverter **INV4** is connected to the input, **106**, of inverter **INV5** and input **A** of **MUX1**. The output, **110**, of inverter **INV5** is connected to the input, **110**, of inverter **INV6**. The output, **112**, of inverter **INV6** is connected to the input, **112**, of inverter **INV7**. The output, **114**, of inverter **INV7** is connected to the input, **114**, of inverter **INV8**. The output, **116**, of inverter **INV8** is connected to the input, **116**, of inverter **INV9** and input **B** of **MUX1**. The output, **118**, of inverter **INV9** is connected to the input, **118**, of inverter **INV10**. The output, **120**, of inverter **INV10** is

connected to the input, **120**, of inverter **INV11**. The output, **122**, of inverter **INV11** is connected to the input, **122**, of inverter **INV12**. The output, **124**, of inverter **INV12** is connected to input **C** of **MUX1**. Control input, **126**, selects either input **A**, **B**, or **C** of **MUX1**.

[015] The output, **128**, of multiplexer **MUX1** is connected to the input, **128**, of inverter **INV13**. The output, **130**, of inverter **INV13** is connected to the input, **130**, of inverter **INV14** and the input, **D**, of multiplexer, **MUX2**. The output, **132**, of inverter **INV14** is connected to the input, **132**, of inverter **INV15** and the input, **C**, of multiplexer, **MUX2**. The output, **134**, of inverter **INV15** is connected to the input, **134**, of inverter **INV16** and the input, **B**, of multiplexer, **MUX2**. The output, **136**, of inverter **INV16** is connected to the input, **A**, of multiplexer, **MUX2**. The output of the programmable delay line is the output, **140**, of the multiplexer, **MUX2**. Control input, **138**, selects either input **A**, **B**, **C**, or **D** of **MUX2**.

[016] The output, **140**, of the multiplexer, **MUX2**, is connected to drains of NFETs, **MN1**, **MN2**, and **MN3**. Control input, **142**, is connected to the gate of **MN1**. Control input, **144**, is connected to the gate of **MN2**. Control input, **146**, is connected to the gate of **MN3**. The source of NFET, **MN1** is connected to a node of capacitor, **CAP1**. The other node of capacitor, **CAP1**, is connected to **GND**. The source of NFET, **MN2** is connected to a node of capacitor, **CAP2**. The other node of capacitor, **CAP2**, is connected to **GND**. The source of NFET, **MN3** is connected to a node of capacitor, **CAP3**. The other node of capacitor, **CAP3**, is connected to **GND**.

[017] A course adjustment of the programmable delay shown in Figure 1 may be achieved by either selecting input **A**, **B**, or **C** of multiplexer, **MUX1**. A delay of four inverters may be achieved by selecting input **A**. A delay of eight inverters may

be achieved by selecting input **B**. A delay of twelve inverters may be achieved by selecting input **C**.

[018] A fine adjustment of the programmable delay shown in Figure 1 may be achieved by either selecting input **A**, **B**, **C**, or **D** of multiplexer, **MUX2**. A delay of four inverters may be achieved by selecting input **A**. A delay of three inverters may be achieved by selecting input **B**. A delay of two inverters may be achieved by selecting input **C**. A delay of one inverter may be achieved by selecting input **D**.

[019] The values of capacitors, **CAP1**, **CAP2**, and **CAP3** are selected to be binary values of each other. For example, if the value of **CAP1** is 1pF, **CAP2** could be 2pF and **CAP3** could 4pF. In this way, by selecting specific combinations of control signals **142**, **144**, and **146**, the available load values on node **140** are 1pF, 2pF, 3pF, 4pF, 5pF, 6pF, and 7pF. By varying the load on node **140**, the delay of the programmable delay line may be adjusted with a resolution finer than the resolution required to measure on-chip, cycle-to-cycle jitter.

[020] By using different combinations of control inputs **126**, **128**, **142**, **144**, and **146** the delay through the programmable delay line shown in Figure 1 may be adjusted as needed. Figure 1 is only an embodiment of a programmable delay line.

[021] Figure 2 is a block diagram of a circuit for measuring on-chip, cycle-to-cycle jitter. A clock signal, **200** is connected to the input of the programmable delay line, **202**, and to an input, of the programmable phase comparator, **208**. The delay in the programmable phase comparator is adjusted by control signals, **204**, such that the clock signal, **200** is delayed one clock cycle. The delayed clock signal, **206** is connected to an input of the programmable phase comparator, **208**.

[022] The programmable phase comparator, **208**, compares the period of clock signal, **200**, to the period of the delayed clock signal, **206**. If the time difference

between the period of the clock signal, **200** and the period of the delayed clock signal, **206** is within the programmed "dead zone" of the programmable phase comparator, **208**, no counter is triggered. The dead zone is a specific time period that is programmed by control signal **210**.

[023] If the difference between the period of the clock signal, **200** and the period of the delayed clock, **206** is greater than the time determined by the dead zone, an output, **212**, from the programmable phase comparator, **208**, triggers counter1, **216**. If the difference between the period of the clock signal, **200**, and the period of the delayed clock, **206**, is negative and its absolute value is greater than the time determined by the dead zone, an output, **214**, from the programmable phase comparator, **208**, triggers counter2, **218**. By setting the dead zone to different values and rerunning the measurement multiple times, a statistical distribution of clock cycle variation may be extracted from the data output, **220**, from counter1, **216**, and from the data output, **222**, from counter2, **218**.

[024] Figure 3 is a drawing illustrating how individual jitter measuring circuits may be placed at different locations on a microprocessor. An example of a clock distribution, **302**, is shown on a microprocessor, **300**. Cycle-to-cycle jitter measurement circuits, J1, **304**, J2, **306**, J3, **308**, J4, **310**, J5, **312**, J6, **314**, J7, **316**, and J8, **318** are placed at various locations on the microprocessor. By placing these circuits at these locations, cycle-to-cycle jitter may be more accurately measured. The jitter measuring circuits are not drawn to scale. Figure 3 is only an illustration.

[025] Figure 4 is a drawing showing distributions of on-chip cycle-to-cycle jitter outside the dead zone at various programmed values of the dead zone. For example, with a the dead zone programmed for 10ps, a certain number of samples, **400**, fall above the dead zone, **DZT1**, and a certain number of samples, **402**, fall

below the dead zone, **DZT1**. By reducing the dead zone time to 5ps, more samples fall outside the dead zone, **DZT2** than when the dead zone was programmed for 10ps, **DZT1**. A certain number of samples, **404**, fall above the dead zone, **DZT2**, and a certain number of samples, **406**, fall below the dead zone, **DZT2**.

[026] By reducing the dead zone time to 2.5ps, more samples fall outside the dead zone, **DZT3** than when the dead zone was programmed for 5ps, **DZT2**. A certain number of samples, **408**, fall above the dead zone, **DZT3**, and a certain number of samples, **410**, fall below the dead zone, **DZT3**. By reducing the dead zone time to 1ps, more samples fall outside the dead zone, **DZT4** than when the dead zone was programmed for 2.5ps, **DZT3**. A certain number of samples, **412**, fall above the dead zone, **DZT4**, and a certain number of samples, **414**, fall below the dead zone, **DZT4**.

[027] Figure 5 is a statistical distribution illustrating how jitter measurement made on-chip may be used to predict cycle-to-cycle jitter. The data obtained in Figure 4 by measuring on-chip jitter many times is used in Figure 5 to determine a statistical distribution.

[028] Distribution **500** and **502** represent the number of samples, **400** and **402**, respectively, measured when the dead zone, **DZT1**, is programmed for 10ps. Distribution **504** and **506** represent the number of samples, **404** and **406**, respectively, measured when the dead zone, **DZT2**, is programmed for 5ps. Distribution **508** and **510** represent the number of samples, **408** and **410**, respectively, measured when the dead zone, **DZT3**, is programmed for 2.5ps. Distribution **512** and **514** represent the number of samples, **412** and **414**, respectively, measured when the dead zone, **DZT4**, is programmed for 1ps.

[029] The distribution, **516**, represented by samples **500**, **504**, **508**, **512**, **514**, **510**, **506**, and **502** may be considered a gaussian distribution. When a distribution, **516**, is gaussian, the percentage of on-chip cycle-to-cycle jitter falling outside a mean on-chip, cycle-to-cycle jitter may be regularly predicted. Knowing these percentages allows a microprocessor designer to plan clock budgets as well as make changes in clocks and clock distributions for future designs.

[030] The foregoing description of the present invention has been presented for purposes of illustration and description. It is not intended to be exhaustive or to limit the invention to the precise form disclosed, and other modifications and variations may be possible in light of the above teachings. The embodiment was chosen and described in order to best explain the principles of the invention and its practical application to thereby enable others skilled in the art to best utilize the invention in various embodiments and various modifications as are suited to the particular use contemplated. It is intended that the appended claims be construed to include other alternative embodiments of the invention except insofar as limited by the prior art.